

**Amendments to the Claims:**

This listing of claims will replace all prior versions, and listings, of claims in the application:

**Listing of Claims:**

1. (Currently Amended) A program development apparatus used for developing a program to be installed in a system having at least a first central processing unit and an other component, said program development apparatus comprising:

a program generating section for generating said program and an event pseudo-generating routine for pseudo-generating said an event based on a state-transition matrix and event pseudo-generating information for pseudo-generating a same event as an event which normally occurs based on data or a signal transmitted from said other component to said first central processing unit in said system, wherein said state-transition matrix has a plurality of cells, each of said cells defined by: a) a state in which said system to be a subject of a program development is enabled to be operate in, and b) an event which is corresponds to an impulse from outside or inside of input to said system, and further wherein a content of information corresponding to a process to be executed by said system and a state of a transition state destination to be transited to when a corresponding event occurs under during a corresponding state are described in stored for each said cell;

a second central processing unit having a same function as said first central processing unit and for executing emulation of said program and said event pseudo-generating routine; and

an analysis section for starting said emulation of said program from a state input as an initial state and for referring to said pseudo-generating information and rewriting information for pseudo-generating said event memorized stored in a memory section used in executing said event pseudo-generating routine into information corresponding to said event which is instructed to occur.

2. (Currently Amended) A program development apparatus used for developing a program to be installed in a system having at least a first central processing unit and an other component, said program development apparatus comprising:

a state-transition matrix memory section for ~~memorizing~~ storing a state-transition matrix, wherein said state-transition matrix has a plurality of cells, each of said cells defined by: a) a state in which said system to be a subject of a program development is enabled to be operate in, and b) an event which corresponds to ~~is an impulse from an outside or an inside of~~ input to said system and further wherein ~~a content of~~ information corresponding to a process to be executed by said system and a ~~state of a transition~~ state destination to be transited to when a corresponding event occurs ~~under~~ during a corresponding state are ~~described in~~ stored for each said cell;

an event pseudo-generating editor for generating event pseudo-generating information for pseudo-generating a same event as an event which normally occurs based on data or a signal transmitted from said other component to a first central processing unit in said system;

a program generating section for generating said program and an event pseudo-generating routine for pseudo-generating said event;

a second central processing unit for having a same function as said first central processing unit and for executing emulation of said program and said event pseudo-generating routine;

an input section for detecting which display position of each event or each state is indicated among a plurality of events and a plurality of states forming said state-transition matrix displayed on a display section and for outputting position information of said display position; and

an analysis section for converting said position information into an event code or a state code corresponding to said position so as to set a state corresponding to said state code as an initial state for starting emulation of said program and for referring to said pseudo-generating information so as to rewrite information ~~memorized~~ stored in a memory section used in executing said

pseudo-generating routine, said information for pseudo-generating an event into information corresponding to said event code.

3. (Currently Amended) A program development apparatus used for developing a program to be installed in a system having at least a first central processing unit and an other component, said program development apparatus comprising:

a state-transition matrix memory section for ~~memorizing~~ storing a state-transition matrix, wherein said state-transition matrix has a plurality of cells, each of said cells defined by: a) a state in which said system to be a subject of a program development is enabled to be operate in, and b) an event which corresponds to is an impulse from an outside or an inside of input to said system and further wherein ~~a content of~~ information corresponding to a process to be executed by said system and a ~~state of a~~ transition state destination to be transited to when a corresponding event occurs ~~under~~ during a corresponding state are ~~described in~~ stored for each said cell;

an event pseudo-generating editor for generating event pseudo-generating information for pseudo-generating a same event as an event which normally occurs based on data or a signal transmitted from said other component to a first central processing unit in said system;

a program generating section for generating said program and an event pseudo-generating routine for pseudo-generating said event;

a second central processing unit for having a same function as said first central processing unit and for executing emulation of said program and said event pseudo-generating routine;

an input section for detecting which display position of each event or each state is indicated among a plurality of events and a plurality of states forming said state-transition matrix displayed on a display section so as to output position information of said display position and for generating an input event log including an order of instructed events and an instruction timing of each event; and

Application No. 09/771,718

Attorney Docket No. 045054/0135

a script generating section for generating a script file in which an occurrence timing of each event and a timing at which an element in said system operates in accordance with a specification are described based on said input event log;

a script analysis section for sequentially outputting position information of each event described in said script file and of a corresponding display area in said state-transition matrix displayed on said display section in order and at an occurrence timing described in said script file; and

an analysis section for converting said position information into an event code or a state code corresponding to said position so as to set a state corresponding to said state code as an initial state for starting emulation of said program and for referring to said pseudo-generating information so as to rewrite information memorized in a memory section used in executing said pseudo-generating routine, said information for pseudo-generating an event into information corresponding to said event code.

4. (Original) The program development apparatus according to Claim 3 further comprising:

a script editor for editing said script file based on any one of an event input to be occurred, an occurrence timing of said event and an occurrence frequency.

5. (Original) The program development apparatus according to Claim 3, wherein said script file is any one of a timing chart format, a text format and a message sequence chart format.

6. (Original) The program development apparatus according to claim 1, wherein said program includes a main routine for executing a main process of said system and a normal generating event routine for normally generating a corresponding event based on various data and a signal transmitted from said other component to said first central processing unit.

7. (Original) The program development apparatus according to Claim 2, wherein said program includes a main routine for executing a main process of said system and a normal generating event routine for normally generating a corresponding event based on various data and a signal transmitted from said other component to said first central processing unit.

8. (Original) The program development apparatus according to Claim 3, wherein said program includes a main routine for executing a main process of said system and a normal generating event routine for normally generating a corresponding event based on various data and a signal transmitted from said other component to said first central processing unit.

9. (Currently Amended) The program development apparatus according to Claim 1, wherein said event pseudo-generating information is information of a generating technique in accordance with said event,

wherein said program includes: a) a main routine for executing main processes of the system, and b) an event normal generating routine for, based on a signal supplied from the system due to operator selection performed on the system, detecting the operator selection and notifying said main routine of the operator selection, and

wherein said pseudo-generating routine is automatically generated in a programming language that is the same as or similar to a programming language of said main routine.

10. (Currently Amended) The program development apparatus according to Claim 2, wherein said event pseudo-generating information is information of a generating technique in accordance with said event,

wherein said program includes: a) a main routine for executing main processes of the system, and b) an event normal generating routine for, based on a signal supplied from the system due to operator selection performed on the

system, detecting the operator selection and notifying said main routine of the operator selection, and

wherein said pseudo-generating routine is automatically generated in a programming language that is the same as or similar to a programming language of said main routine.

11. (Currently Amended) The program development apparatus according to Claim 3, wherein said event pseudo-generating information is information of a generating technique in accordance with said event,

wherein said program includes: a) a main routine for executing main processes of the system, and b) an event normal generating routine for, based on a signal supplied from the system due to operator selection performed on the system, detecting the operator selection and notifying said main routine of the operator selection, and

wherein said pseudo-generating routine is automatically generated in a programming language that is the same as or similar to a programming language of said main routine.

12. (Original) The program development apparatus according to Claim 1, wherein said event is any one of a message-type for receiving a start message from another task or another apparatus, a flag-type for reading a variation of a variable or an input/output, an interrupt-type for receiving an interrupt from an outside, an in-mail type for notifying an internal event which occurs in a cell of said state-transition matrix to another state-transition matrix when said state-transition matrix is layered and a function-call type for calling a function executing a group of processes.

13. (Original) The program development apparatus according to Claim 2, wherein said event is any one of a message-type for receiving a start message from another task or another apparatus, a flag-type for reading a

variation of a variable or an input/output, an interrupt-type for receiving an interrupt from an outside, an matrix when said state-transition matrix is layered and a function-call type for calling a function executing a group of processes.

14. (Original) The program development apparatus according to Claim 3, wherein said event is any one of a message-type for receiving a start message from another task or another apparatus, a flag-type for reading a variation of a variable or an input/output, an interrupt-type for receiving an interrupt from an outside, an in-mail type for notifying an internal event which occurs in a cell of said state-transition matrix to another state-transition matrix when said state-transition matrix is layered and a function-call type for calling a function executing a group of processes.

15. (Currently Amended) A program development method used for developing a program to be installed in a system having at least a first central processing unit and an other component, said program development method comprising:

a first step of generating said program and an event pseudo-generating routine for pseudo-generating said an event based on a state-transition matrix and event pseudo-generating information for pseudo-generating a same event as an event which normally occurs based on data or a signal transmitted from said other component to said first central processing unit in said system, wherein said state-transition matrix has a plurality of cells, each of said cells defined by: a) a state in which said system to be a subject of a program development is enabled to be operate in, and b) an event which corresponds to an impulse from outside or inside of input to said system, and further wherein ~~a content of information corresponding to~~ a process to be executed by said system and a ~~state of a transition state~~ destination to be transited to when a corresponding event occurs ~~under~~ during a corresponding state are ~~described in~~ stored for each said cell; and

a second step of starting emulation of said program from a state input as an initial state, of referring to said pseudo-generating information while executing said event pseudo-generating routine and of rewriting information for pseudo-generating said event ~~memorized~~ stored in a memory section used in executing said event pseudo-generating routine into information corresponding to said event which is instructed to occur.

16. (Currently Amended) A program development method used for developing a program to be installed in a system having at least a first central processing unit and an other component, and carried out by using:

a state-transition matrix memory section for ~~memorizing~~ storing a state-transition matrix, wherein said state-transition matrix has a plurality of cells, each of said cells defined by: a) a state in which said system to be a subject of a program development is enabled to be operate in, and b) an event which corresponds to ~~is an impulse from an outside or an inside of~~ input to said system and further wherein ~~a content of~~ information corresponding to a process to be executed by said system and a ~~state of a~~ transition state destination to be transited to when a corresponding event occurs ~~under~~ during a corresponding state are ~~described in~~ stored for each said cell;

an input section for detecting a display position of which event or state is instructed among a plurality of events or a plurality of states forming said state-transition matrix displayed on a display section and for outputting position information about detected said display position, said program development method comprising:

a first step of generating event pseudo-generating information for pseudo-generating a same event as an event normally generated based on data or a signal transmitted from said other component to a first central processing unit in said system;

a second step of generating said program and an event pseudo-generating routine for pseudo-generating said event based on said state-transition matrix and said event pseudo-generating information; and



a third step of converting said position information into an event code or a state code corresponding to said position, of starting emulation of said program from a state input as an initial state, of referring to said pseudo-generating information while executing said event pseudo-generating routine and of rewriting information for pseudo-generating said event ~~memorized~~ stored in a memory section used in executing said event pseudo-generating routine into information corresponding to said event which is instructed to occur.

17. (Currently Amended) A program development method used for developing a program to be installed in a system having at least a first central processing and an other component, and carried out by using:

a state-transition matrix memory section for ~~memorizing~~ storing a state-transition matrix, wherein said state-transition matrix has a plurality of cells, each of said cells defined by: a) a state in which said system to be a subject of a program development is enabled to be operate in, and b) an event which corresponds to is an impulse from an outside or an inside of input to said system and further wherein ~~a content of~~ information corresponding to a process to be executed by said system and a ~~state of a transition~~ state destination to be transited to when a corresponding event occurs ~~under~~ during a corresponding state are ~~described in~~ stored for each said cell;

an input section for detecting a display position of which event or state is instructed among a plurality of events or a plurality of states forming said state-transition matrix displayed on a display section and for outputting position information about detected said display position, said program development method comprising:

a first step of generating event pseudo-generating information for pseudo-generating a same event as an event normally generated based on data or a signal transmitted from said other component to a first central processing unit in said system;

a second step of generating said program and an event pseudo-generating routine for pseudo-generating said event based on said state-transition matrix and said event pseudo-generating information;

a third step of generating an input event log including an order of instructed events and a timing at which each event is instructed;

a fourth step, based on said input event log, of generating a script file in which an occurrence timing of each event described in said state-transition matrix and a timing at which an element in said system operates in accordance with a specification are described;

a fifth step of sequentially outputting position information of each event described in said script file and of a corresponding display area in said state-transition matrix displayed on said display section in order and at an occurrence timing described in said script file; and

a sixth step of converting said position information into an event code corresponding to said position, of referring to said event pseudo-generating information while executing said event pseudo-generating routine and of rewriting information ~~memorized~~ stored in a memory section used by said event pseudo-generating routine, said information for pseudo-generating an event into information corresponding to said event code.

18. (Original) The program development method according to Claim 17, further comprising:

a seventh step of editing said script file based on any one of an event input to be occurred, an occurrence timing of said event and an occurrence frequency.

19. (Original) The program development method according to Claim 17, wherein said script file is any one of a timing chart format, a text format and a message sequence chart format.

20. (Original) The program development method according to Claim 15 wherein said program includes a main routine for executing a main process of said system and a normal generating event routine for normally generating a corresponding event based on various data and a signal transmitted from said other component to said first central processing unit.

21. (Original) The program development method according to Claim 16 wherein said program includes a main routine for executing a main process of said system and a normal generating event routine for normally generating a corresponding event based on various data and a signal transmitted from said other component to said first central processing unit.

22. (Original) The program development method according to Claim 17 wherein said program includes a main routine for executing a main process of said system and a normal generating event routine for normally generating a corresponding event based on various data and a signal transmitted from said other component to said first central processing unit.

23. (Original) The program development method according to Claim 15, wherein said event pseudo-generating information is information of a generating technique in accordance with said event.

24. (Original) The program development method according to Claim 16, wherein said event pseudo-generating information is information of a generating technique in accordance with said event.

25. (Original) The program development method according to Claim 17, wherein said event pseudo-generating information is information of a generating technique in accordance with said event.

26. (Original) The program development method according to Claim 15, wherein said event is any one of a message-type for receiving a start message from another task or another apparatus, a flag-type for reading a variation of a variable or an input/output, an interrupt-type for receiving an interrupt from an outside, an in-mail type for notifying an internal event which occurs in a cell of said state-transition matrix to another state-transition matrix when said state-transition matrix is layered and a function-call type for calling a function executing a group of processes.

27. (Original) The program development method according to Claim 16, wherein said event is any one of a message-type for receiving a start message from another task or another apparatus, a flag-type for reading a variation of a variable or an input/output, an interrupt-type for receiving an interrupt from an outside, an in-mail type for notifying an internal event which occurs in a cell of said state-transition matrix to another state-transition matrix when said state-transition matrix is layered and a function-call type for calling a function executing a group of processes.

28. (Original) The program development method according to Claim 17, wherein said event is any one of a message-type for receiving a start message from another task or another apparatus, a flag-type for reading a variation of a variable or an input/output, an interrupt-type for receiving an interrupt from an outside, an in-mail type for notifying an internal event which occurs in a cell of said state-transition matrix to another state-transition matrix when said state-transition matrix is layered and a function-call type for calling a function executing a group of processes.

29. (Currently Amended) A program development program for causing a computer to carry out a program development method used for developing a program to be installed in a system having at least a first central

processing unit and an other component, said program development method comprising:

a first step of generating said program and an event pseudo-generating routine for pseudo-generating said an event based on a state-transition matrix and event pseudo-generating information for pseudo-generating a same event as an event which normally occurs based on data or a signal transmitted from said other component to said first central processing unit in said system, wherein said state-transition matrix has a plurality of cells, each of said cells defined by: a) a state in which said system to be a subject of a program development is enabled to be operate in, and b) an event which is corresponds to an impulse from outside or inside of input to said system, and further wherein ~~a content of information corresponding to~~ a process to be executed by said system and a ~~state of a transition state~~ destination to be transited to when a corresponding event occurs ~~under~~ during a corresponding state are ~~described in~~ stored for each said cell; and

a second step of starting emulation of said program from a state input as an initial state, of referring to said pseudo-generating information while executing said event pseudo-generating routine and of rewriting information for pseudo-generating said event ~~memorized~~ stored in a memory section used in executing said event pseudo-generating routine into information corresponding to said event which is instructed to occur.

30. (Currently Amended) A storage medium storing a program development program for causing a computer to carry out a program development method used for developing a program to be installed in a system having at least a first central processing unit and an other component, said program development method comprising:

a first step of generating said program and an event pseudo-generating routine for pseudo-generating said an event based on a state-transition matrix and event pseudo-generating information for pseudo-generating a same event as an event which normally occurs based on data or a signal transmitted from said

other component to said first central processing unit in said system, wherein said state-transition matrix has a plurality of cells, each of said cells defined by: a) a state in which said system to be a subject of a program development is enabled to be operate in, and b) an event which is corresponds to an impulse from outside or inside of input to said system, and further wherein ~~a content of information corresponding to~~ a process to be executed by said system and a ~~state of a transition state~~ destination to be transited to when a corresponding event occurs ~~under~~ during a corresponding state are ~~described in~~ stored for each said cell; and

a second step of starting emulation of said program from a state input as an initial state, of referring to said pseudo-generating information while executing said event pseudo-generating routine and of rewriting information for pseudo-generating said event ~~memorized~~ stored in a memory section used in executing said event pseudo-generating routine into information corresponding to said event which is instructed to occur.

31. (Currently Amended) A program development program for causing a computer to carry out a program development program for causing a computer to carry out a program development method used for developing a program to be installed in a system having at least a first central processing and an other component, said program development method comprising:

a state-transition matrix memory section for ~~memorizing~~ storing a state-transition matrix, wherein said state-transition matrix has a plurality of cells, each of said cells defined by: a) a state in which said system to be a subject of a program development is enabled to be operate in, and b) an event which corresponds to is an impulse from an outside or an inside of input to said system and further wherein ~~a content of~~ information corresponding to a process to be executed by said system and a ~~state of a transition state~~ destination to be transited to when a corresponding event occurs ~~under~~ during a corresponding state are ~~described in~~ stored for each said cell;

an input section for detecting a display position of which event or state is instructed among a plurality of events or a plurality of states forming said state-transition matrix displayed on a display section and for outputting position information about detected said display position;

a first step of generating event pseudo-generating information for pseudo-generating a same event as an event normally generated based on data or a signal transmitted from said other component to a first central processing unit in said system;

a second step of generating said program and an event pseudo-generating routine for pseudo-generating said event based on said state-transition matrix and said event pseudo-generating information; and

a third step of converting said position information into an event code or a state code corresponding to said position, of starting emulation of said program from a state input as an initial state, of referring to said pseudo-generating information while executing said event pseudo-generating routine and of rewriting information for pseudo-generating said event ~~memorized~~ stored in a memory section used in executing said event pseudo-generating routine into information corresponding to said event which is instructed to occur.